## Classification of Digital Systems

- 1) Linear and Nonlinear Digital Systems
- 2) Time-Invariant and Time-Varying Digital Systems
- 3) Causal and Non-Causal Digital Systems
- 4) Stable and Unstable Digital Systems




## Linear Digital Systems

- If $y_1[n]$ and $y_2[n]$ are the output sequences of a linear digital system, for input sequences $x_1[n]$ and $x_2[n]$, respectively, then for an input sequence

$$x[n] = \alpha x_1[n] + \beta x_2[n]$$

the output sequence is

$$y[n] = \alpha y_1[n] + \beta y_2[n]$$




## Linear Digital Systems

- These equations must hold for any arbitrary constants $\alpha$ and $\beta$, and for all possible input sequences $x_1[n]$ and $x_2[n]$
- Hence, an infinite number of measurements are required to test the linearity property of a digital system
- The system is a nonlinear digital system if these equations do not hold for one or more constants $\alpha$ and $\beta$, and/or one or more inputs




## Linear Digital Systems

**Example** – Let $y_1[n]$ and $y_2[n]$ denote the outputs of the nonrecursive accumulator for inputs $x_1[n]$ and $x_2[n]$, respectively:

$$y_1[n] = \sum_{\ell=-\infty}^{n} x_1[\ell]$$

$$y_2[n] = \sum_{\ell=-\infty}^{n} x_2[\ell]$$




## Linear Digital Systems

- Then for an input $x[n] = \alpha x_1[n] + \beta x_2[n]$, the output is given by

$$y[n] = \sum_{\ell=-\infty}^{n} (\alpha x_1[\ell] + \beta x_2[\ell])$$

$$= \alpha \sum_{\ell=-\infty}^{n} x_1[\ell] + \beta \sum_{\ell=-\infty}^{n} x_2[\ell] = \alpha y_1[n] + \beta y_2[n]$$

- Thus, the nonrecursive accumulator is a linear digital system




## Linear Digital Systems

- Now consider the recursive accumulator with a causal input sequence starting at $n = 0$
- Here, for input sequences $x_1[n]$ and $x_2[n]$, the output sequences $y_1[n]$ and $y_2[n]$, respectively, are given by

$$y_1[n] = y_1[-1] + \sum_{\ell=0}^{n} x_1[\ell]$$

$$y_2[n] = y_2[-1] + \sum_{\ell=0}^{n} x_1[\ell]$$

## Linear Digital Systems

- Here, for an input $x[n] = \alpha x_1[n] + \beta x_2[n]$, the output is given by

$$y[n] = y[-1] + \sum_{\ell=0}^{n} (\alpha x_1[\ell] + \beta x_2[\ell])$$
$$= y[-1] + \alpha \sum_{\ell=0}^{n} x_1[\ell] + \beta \sum_{\ell=0}^{n} x_2[\ell]$$




## Linear Digital Systems

- On the other hand $\alpha y_1[n] + \beta y_2[n]$

$$= \alpha\left(y_1[-1] + \sum_{\ell=0}^{n} x_1[\ell]\right) + \beta\left(y_2[-1] + \sum_{\ell=0}^{n} x_2[\ell]\right)$$
$$= \alpha y_1[-1] + \alpha \sum_{\ell=0}^{n} x_1[\ell] + \beta y_2[-1] + \beta \sum_{\ell=0}^{n} x_2[\ell]$$

- Hence, $y[n] = \alpha y_1[n] + \beta y_2[n]$ if

$$y[-1] = \alpha y_1[-1] + \beta y_2[-1]$$




## Linear Digital Systems

- The last condition must be satisfied for all possible constants $\alpha$ and $\beta$, and all possible initial conditions $y[-1]$, $y_1[-1]$, and $y_2[-1]$ to ensure that the recursive accumulator be a linear system
- These restrictions cannot be met unless the recursive accumulator is at rest with zero initial condition at the time instant when the input is applied, that is, $y[-1] = 0$




## Nonlinear Digital Systems

- A recursive accumulator with nonzero initial condition is a nonlinear digital system
- Fairly simple nonlinear digital systems have been used for signal enhancement




## Nonlinear Digital Systems

- One such nonlinear system is the twicing function defined by

$$y[n] = x[n] \cdot (2 - x[n])$$

- Another nonlinear system is the Teager operator given by

$$y[n] = x^2[n] - x[n-1] \cdot x[n+1]$$




## Time-Invariant Digital Systems

- If $y[n]$ is the output sequence of a time-invariant digital system for an input sequence $x[n]$, then for a input sequence $x_1[n] = x[n - N_o]$ time-shifted by a positive or negative integer amount $N_o$, the output sequence $y_1[n]$ is also time-shifted by the same amount; that is,

$$y_1[n] = y[n - N_o]$$

## Time-Invariant Digital Systems

- Moreover, the time-invariance property must hold for all possible input sequences
- Thus, for a given input sequence, the output sequence of a time-invariant system remains the same irrespective of the time instant the input is applied




## Time-Invariant Digital Systems

**Example** - Moving-average filter and linear interpolator are time-invariant digital systems

**Example** - Down-sampler and up-sampler are time-varying digital systems




## Causal Digital Systems

- In a causal digital system, the output sample $y[N_o]$ at a time index $n = N_o$ does not depend on input samples for time indices $n > N_o$ and depend only on input samples for time indices $n \le N_o$




## Causal Digital Systems

- Or, in other words, if $y_1[n]$ and $y_1[n]$ denote the output sequences of a causal digital system for input sequences $x_1[n]$ and $x_2[n]$, respectively, then

$$x_1[n] = x_2[n] \text{ for } n \le N_o \text{ implies}$$
$$y_1[n] = y_2[n] \text{ for } n \le N_o$$

- The above definition of the causality property is only for digital systems with the same input and output sampling rates




## Causal Digital Systems

**Example** – Both types of the accumulators and the moving-average filter are causal digital systems

- The factor-of-2 linear interpolator defined by

$$y[n] = x_u[n] + \tfrac{1}{2}(x_u[n-1] + x_u[n+1])$$

is a non-causal digital system




## Causal Digital Systems

- A causal version of the factor-of-2 interpolator is obtained by delaying the generation of the output sample by one sample period

$$y[n] = x_u[n-1] + \tfrac{1}{2}(x_u[n-2] + x_u[n])$$

## Stable Digital Systems

- A digital system is said to be a BIBO stable, if for any bounded input sequence, the corresponding output sequence is also bounded, that is,
  $|x[n]| < C_x < +\infty$ implies
  $$|y[n]| < C_y < +\infty \,, \quad -\infty < n < +\infty$$
  where $C_x$ and $C_y$ are positive constants




## Stable Digital Systems

**Example** – Consider the $M$-point moving-average filter
$$y[n] = \frac{1}{M}\sum_{\ell=0}^{M-1} x[n-\ell]$$

- Here,
  $$|y[n]| = \left|\frac{1}{M}\sum_{\ell=0}^{M-1} x[n-\ell]\right| \le \frac{1}{M}\sum_{\ell=0}^{M-1} |x[n-\ell]|$$




## Stable Digital Systems

- For an input sequence $x[n]$ bounded above by a positive constant $C_x$, we have
  $$|y[n]| \le \frac{1}{M}(M)C_x \le C_x$$
  indicating that $y[n]$ is also a bounded sequence
- Hence, the moving-average filter is a stable digital system




## Impulse Response

- The output of a digital system for an unit sample sequence $\delta[n]$ as the input is known as its impulse response, to be denoted as $h[n]$

**Example** – Consider the ideal delay system given by $y[n] = x[n - N_o]$

- By setting $x[n] = \delta[n]$ in the input-output relation we get
  $$h[n] = \delta[n - N_o]$$




## Impulse Response

**Example** – Consider the nonrecursive accumulator given by
$$y[n] = \sum_{\ell=-\infty}^{n} x[\ell]$$

- By setting $x[n] = \delta[n]$ in the above equation we have
  $$h[n] = \sum_{\ell=-\infty}^{n} \delta[\ell] = \mu[n]$$




## Impulse Response

**Example** – Consider the factor-of-2 linear interpolator given by
$$y[n] = x_u[n] + \tfrac{1}{2}(x_u[n-1] + x_u[n+1])$$

- By setting $x_u[n] = \delta[n]$ in the above equation we have
  $$h[n] = \delta[n] + \tfrac{1}{2}(\delta[n-1] + \delta[n+1])$$
- Hence,
  $$\{h[n]\} = \{0.5,\ 1,\ 0.5\}, \quad -1 \le n \le 1$$

## LTI Digital System

- In this course we shall be almost exclusively concerned with the linear, time-invariant (LTI) digital system that satisfies both linearity and time-invariance properties
- This class of systems are simpler to characterize and design
- In addition we shall also impose the causality and stability properties on the LTI systems of interest




## Time-Domain Representation of a Sequence

- An arbitrary sequence $\{x[n]\}$ as a linear, weighted sum of the delayed and advanced unit sample sequences
- To illustrate this representation, consider first the finite-length sequence $\{x[n]\} = \{a_{-1},\ 0,\ a_1,\ 0,\ a_1\}$, $-1 \le n \le 3$




## Time-Domain Representation of a Sequence

- The $n$-th sample of this sequence can be written as
  $x[n] = a_{-1}\delta[n+1] + a_1\delta[n-1] + a_3\delta[n-3]$
- To verify this representation we note that for $n < -1$ and $n > 3$, $x[n] = 0$
- Now,
  $x[-1] = a_{-1}\delta[0] + a_1\delta[-2] + a_3\delta[-4] = a_{-1}$
  $x[0] = a_{-1}\delta[1] + a_1\delta[-1] + a_3\delta[-3] = 0$




## Time-Domain Representation of a Sequence

$x[1] = a_{-1}\delta[2] + a_1\delta[0] + a_3\delta[-2] = a_1$
$x[2] = a_{-1}\delta[3] + a_1\delta[1] + a_3\delta[-1] = 0$
$x[3] = a_{-1}\delta[4] + a_1\delta[2] + a_3\delta[0] = a_3$

- Generalizing the above result, we conclude that an arbitrary sequence $\{x[n]\}$ can be expressed as

$$x[n] = \sum_{\ell=-\infty}^{\infty} x[\ell]\delta[n-\ell]$$




## Time-Domain Representation of a Sequence

- Note: The term $x[\ell]\delta[n-\ell]$ on the RHS is an unit sample sequence of weight $x[\ell]$ located at $n = \ell$
- Also, $x[\ell]$ is the amplitude of the $\ell$-th sample of the sequence $\{x[n]\}$




## Convolution Sum

- Let $h[n]$ denote the impulse response of an LTI digital system, which is the output of the system for an input $\delta[n]$
- We determine its output for an input
  $x[n] = a_{-1}\delta[n+1] + a_1\delta[n-1] + a_3\delta[n-3]$

## Convolution Sum

- Since the system is linear and time-invariant, the output for an input $a_{-1}\delta[n+1]$ is $a_{-1}h[n+1]$
- Likewise, the output for an input $a_1\delta[n-1]$ is $a_1h[n-1]$, and the output for an input $a_3\delta[n-3]$ is $a_3h[n-3]$

31



## Convolution Sum

- Therefore, it follows that because of linearity property the output for an input $x[n]=a_{-1}\delta[n+1]+a_1\delta[n-1]+a_3\delta[n-3]$ is given by $y[n]=a_{-1}h[n+1]+a_1h[n-1]+a_3h[n-3]$

32



## Convolution Sum

- Now an arbitrary signal $x[n]$ can be expressed as
$$x[n]=\sum_{k=-\infty}^{\infty}x[k]\delta[n-k]$$
- For an LTI digital system with an impulse response $h[n]$, the output for an input $x[k]\delta[n-k]$ is simply $x[k]h[n-k]$

33



## Convolution Sum

- Hence, for an input
$$x[n]=\sum_{k=-\infty}^{\infty}x[k]\delta[n-k]$$
the output is given by
$$\boxed{y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]}$$
- The above equation is known as the convolution sum

34



## Convolution Sum

- The convolution sum is a fundamental property of an LTI digital system
- Knowing the impulse response $h[n]$ we can determine the output $y[n]$ of the LTI system for any arbitrary input $x[n]$ that can be expressed as a linear weighted sum of the unit sample sequence $\delta[n]$ and its delayed and advanced versions using the convolution sum

35



## Convolution Sum

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]$$

- Note: Sum of the indices of all terms inside the summation is equal to the index on the left-hand side
- Compact form: $y[n]=x[n]\circledast h[n]$

36

## Convolution Sum

**Example** – The convolution sum a signal $x[n]$, the output $y[n]$ of a LTI system is given by

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k] = x[n]$$

**Convolution Sum of Two Causal Sequences**

- The convolution sum operation is carried over the dummy integer variable $k$, and not over the time index $n$

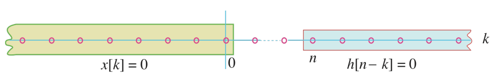



## Convolution Sum

- If $x[n]$ is a causal sequence, then $x[k] = 0$ for $-\infty < k < 0$
- Similarly, if $h[n]$ is a causal sequence, then $h[n-k] = 0$ for $n-k < 0$, or, equivalently, for $n < k < +\infty$
- The ranges of the time index $k$ for which $x[k] = 0$ and $h[n-k] = 0$ are shown by the lightly shaded regions in the figure on next slide



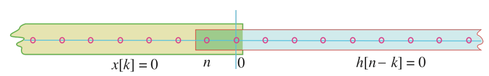

## Convolution Sum




- Consequently, the product $x[k]h[n-k] = 0$ for $k < 0$ and $k > n$, and the range of the dummy variable $k$ of the convolution sum is from $k = 0$ to $k = n$
- On the other hand, if $n < 0$, the product $x[k]h[n-k] = 0$ for all values of the dummy variable $k$ as shown by the lightly shaded regions in the figure in the next slide




## Convolution Sum




- As a result, when both $x[n]$ and $h[n]$ are causal sequences, then the convolution sum reduces to

$$y[n] = \begin{cases} \sum_{k=0}^{n} x[k]h[n-k], & n \geq 0 \\ 0, & n < 0 \end{cases}$$




## Convolution Sum

**Convolution Sum of Two Finite-Length Sequences**

- The convolution sum $y[n]$ of two finite-length sequences, $x[n]$ and $h[n]$ is also finite-length
- Let $x[n]$ be of length-$L$ and defined for $N_1 \leq n \leq N_2$
- Let $h[n]$ be of length-$M$ and defined for $N_3 \leq n \leq N_4$




## Convolution Sum

- Note: $L = N_2 - N_1 + 1$ and $M = N_4 - N_3 + 1$
- The first non-zero sample of $y[n]$ is $y[N_1 + N_3] = x[N_1]h[N_3]$ and the last non-zero sample is $y[N_2 + N_4] = x[N_2]h[N_4]$
- Hence, $y[n]$ is a finite-length sequence of length

$$\begin{aligned} &(N_2 + N_4) - (N_1 + N_3) + 1 \\ &= (N_2 - N_1 + 1) + (N_4 - N_3 + 1) + 1 \\ &= L + M - 1 \end{aligned}$$

## Convolution Sum

**Example** – We determine $y[n] = g[n] \circledast h[n]$ where $g[n]$ is a length-3 Sequence defined for $0 \le n \le 2$ and $h[n]$ is a length-4 sequence defined for $0 \le n \le 3$

- Length of $y[n]$ is $3 + 4 - 1 = 6$
- The 6 samples of $y[n]$, $0 \le n \le 5$, are given in the next slide

43



## Convolution Sum

$$y[0] = g[0]h[0]$$
$$y[1] = g[0]h[1] + g[1]h[0]$$
$$y[2] = g[0]h[2] + g[1]h[1] + g[2]h[0]$$
$$y[3] = g[0]h[3] + g[1]h[2] + g[2]h[1]$$
$$y[4] = g[1]h[3] + g[2]h[2]$$
$$y[5] = g[2]h[3]$$

- The index of the sequence $y[n]$ for each value of $n$ is precisely the sum of the indices of each product $g[k]h[n-k]$ on the RHS

44



## Convolution Sum

**Example** – Let $y[n] = x[n] \circledast h[n]$ where

$$x[n] = \{3,\ 5,\ -4,\ 1,\ -2\},\ 0 \le n \le 4$$
$$h[n] = \{2,\ -3,\ 1,\ 5\},\ 0 \le n \le 3$$

- Determine $y[4]$ without computing all samples of the convolution sum
- To determine $y[4]$ we choose the product terms whose sum of indices are 4

45



## Convolution Sum

- Hence,

$$y[4] = x[1]h[3] + x[2]h[2] + x[3]h[1] + x[4]h[0]$$
$$= 5 \times 5 + (-4) \times 1 + 1 \times (-3) + (-2) \times 2$$
$$= 25 - 4 - 3 - 4 = 14$$

46



## Graphical Interpretation of Convolution Sum

- Consider the two length-6 sequences

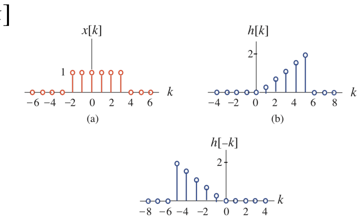
$$x[n] = \mu[n+2] - \mu[n-2]$$
$$= \{1,\ 1,\ 1,\ 1,\ 1,\ 1\},\ -2 \le n \le 3$$
$$h[n] = 0.4n(\mu[n] - \mu[n-5])$$
$$= 0.4n,\ 0 \le n \le 5$$

47



## Graphical Interpretation of Convolution Sum

- These sequences are shown below as a function of the dummy index $k$ along with $h[-k]$



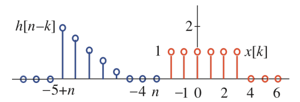

48

## Graphical Interpretation of Convolution Sum

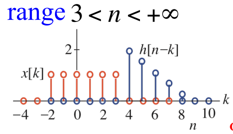
- Next we slide the time-shifted sequence $h[n-k]$ over $x[k]$ for increasing value of $n$ starting at $n = -\infty$
- As can be seen from the figure in the next slide, for $n < -2$, there is no overlap between the samples of $h[n-k]$ and $x[k]$
- As a result, $x[k]h[n-k] = 0$ for all values of the index $n$ in the range $-\infty < n \le -2$




## Graphical Interpretation of Convolution Sum




- Next, it can be seen from the figure below that for $n > 8$, there is also no overlap between $h[n-k]$ and $x[k]$ for for all values of $n$ in the range $3 < n < +\infty$

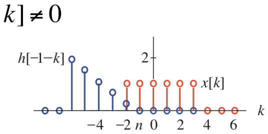






## Graphical Interpretation of Convolution Sum

- There is an overlap between $h[n-k]$ and $x[k]$ for values of $n$ in the range $-2 \le n \le 3$ as shown in the figure below and as a result $x[k]h[n-k] \ne 0$

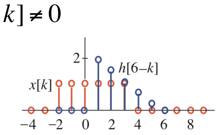






## Graphical Interpretation of Convolution Sum

- Similarly, as shown in the figure below, in the range $3 \le n \le 8$, there is also an overlap between $h[n-k]$ and $x[k]$, and hence, $x[k]h[n-k] \ne 0$







## Convolution Sum

- The computed convolution sum $y[n]$ is given by

$$\{y[n]\} = \{0,\ 0.4,\ 1.2,\ 2.4,\ 4.0,\ 6.0,\ 6.0,\ 5.6,\ 4.8,\ 3.6,\ 2\}$$
$$-2 \le n \le 8$$




## Convolution Sum Using MATLAB

- The function `conv` can be used to compute the convolution sum of two finite-length sequences

## Convolution Sum Using MATLAB

- Code fragments to compute the convolution sum of the sequences in Slide No. 47 are

```
n = 0:1:5;
x = ones(1.6);
h = 0.4*n;
y = conv(x,h);
```

55



## Convolution Sum Properties

Commutative Property

$$h[n]\circledast x[n] = x[n]\circledast h[n]$$

Associative Property

$$(x[n]\circledast h[n])\circledast w[n] = x[n]\circledast(h[n]\circledast w[n])$$

Distributive Property

$$w[n]\circledast(h[n]\circledast x[n]) = w[n]\circledast h[n] + w[n]\circledast x[n]$$

56



## Deconvolution

- Process of determining a causal finite-length sequence from its convolution sum with another causal finite-length sequence is known as the deconvolution
- To develop the algorithm for deconvolution, we consider the convolution sum $y[n]$ of two causal sequences $x[n]$ and $h[n]$

57



## Deconvolution

- From Slide No. 40 we have

$$y[n] = \begin{cases} \sum_{k=0}^{n} x[k]h[n-k], & n \geq 0 \\ 0, & n < 0 \end{cases}$$

- It follows from above $y[0] = x[0]h[0]$, and hence

$$x[0] = \frac{y[0]}{h[0]}$$

58



## Deconvolution

- To evaluate $x[n]$ for $n \geq 1$, we rewrite the equation in the previous slide as

$$y[n] = x[n]h[0] + \sum_{k=0}^{n-1} x[k]h[n-k], \quad n \geq 1$$

- From above we have

$$x[n] = \frac{y[n] - \sum_{k=0}^{n-1} x[k]h[n-k]}{h[0]}, \quad n \geq 1$$

59



## Deconvolution

**Example** – Let $y[n] = \{6, -13, 1, 22, -12, -16\}$ denote the convolution sum of $x[n]$ with $h[n] = \{3, -2, -4\}$

- Since $y[n]$ is of length 6 and $h[n]$ is of length 3, length of $x[n]$ is $6 - 3 + 1 = 4$
- The 4 samples of $x[n]$ are obtained using the formula given in the previous slide

60

## Deconvolution

$$x[0] = \frac{y[0]}{h[0]} = \frac{6}{3} = 2$$

$$x[1] = \frac{y[1] - x[0]h[1]}{h[0]} = \frac{-13 - 2\times(-2)}{3} = -3$$

$$x[2] = \frac{y[2] - x[0]h[2] - x[1]h[1]}{h[0]}$$

$$= \frac{1 - 2\times(-4) - (-3)\times(-2)}{3} = 1$$

61



## Deconvolution

$$x[3] = \frac{y[3] - x[1]h[2] - x[2]h[1]}{h[0]}$$

$$= \frac{22 - (-3)\times(-4) - 1\times(-2)}{3} = 4$$

- Hence, $x[n] = \{2, -3, 1, 4\}$

62



## Deconvolution Using MATLAB

- The function `deconv` can be used to implement the deconvolution algorithm
- Code fragments used to perform the deconvolution of the sequences $y[n]$ and $h[n]$ of the Example in Slide No. 60 are given in the next slide

63



## Deconvolution Using MATLAB

```
y = [6  -13  1  22 -12  -16];
h = [3  -2  -4];
[x,r] = deconv(y,h);
```

which yield

```
x =
   2   -3   1   4
r =
   0   0   0   0   0   0
```

64